Class notes – Computing Intermediate

Operating Systems

What is an OS?

The OS is the manager of the computer.

Example of how an OS manages the computer:

1. The user double clicks on the word processing (WP) icon.
2. The OS orders the loader (L) to copy WP from the hard disk (HD) to the RAM (L is told where WP is found on HD and where HP should be put in RAM).
3. User writes a letter (L) and clicks on the 'print' icon.
4. WP informs the OS that the user wants to print.
5. OS checks whether the printer (P) is busy doing something else. If P is busy it puts L in a queue. If not it communicates with P to print L.
6. User clicks the icon to save L. WP informs OS that WP wants to save L.
7. OS calls a program to save L.
8. User closes WP. OS marks the space in RAM occupied by WP as free.

Types of OS

Batch OS

- Executing a sequence of non-interactive jobs sequentially.
- Today such programs are executed in the background.
- Uses JCL (Job Control Language) e.g. to indicate the files that must be inputted with the job.

Multi-tasking OS

- Several applications simultaneously loaded and used in memory
- Processor switches between applications
- Very common
- Two kinds:
    - Pre-emptive multitasking: processor use is regulated by OS
    - Collaborative multitasking: it is the processes that decide when they stop using the processor.
- Used to be called time-sharing OS.

Real time OS

- Immediate replies (time-limit set between input and output)
- Two types:
    - Hard real-time: time limit has to be respected e.g. nuclear plant, autopilot etc.
    - Soft real-time: time limit can be eased e.g. airline reservations

Network OS

- Designed for a server.
- Manages concurrent requests from clients.
- Provides the security necessary in a multiuser environment.
- Coordinates sharing of files, applications and devices.

Online OS

- Runs on a server that is accessible to the Internet.
- Coordinates online accesses to an application on a network e.g. more than one user are using a word processor.
- User files, emails etc. are stored online.

Single-user OS

- can be split into two types:
    - single user, single application operating systems e.g. mobile phone
    - single user, multi-tasking operating systems e.g. personal computer

Multi-user OS

- More than one user is logged on and can use the computer at the same time.
- Each user runs more than one application at a time, so it needs to be multi-tasking as well.
- Also called multi-access.

Multiprocessing OS

- Supports the running of a program on more than one CPU.

Multithreading OS

- Allows different parts of a single program to run concurrently.

Interactive OS

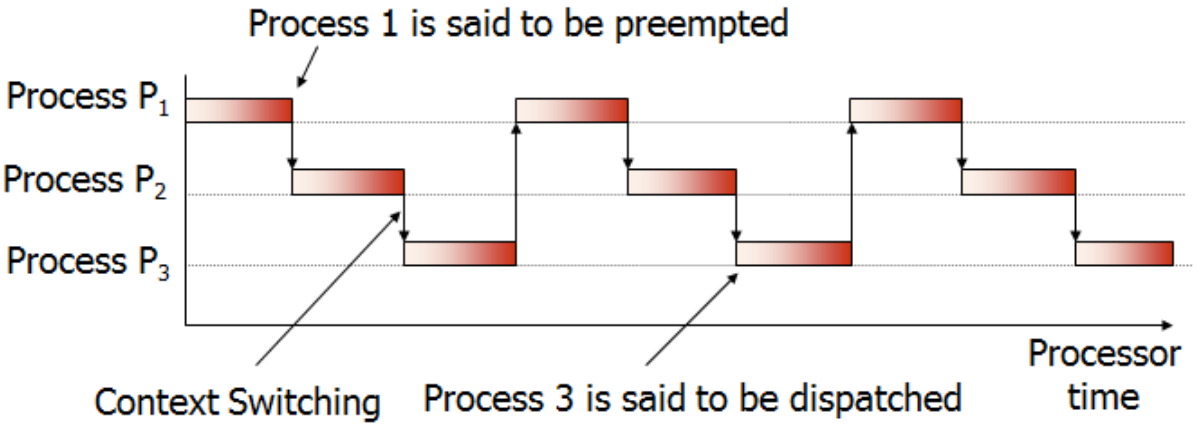
- Allows programs that interact with the user e.g. to play games.

The main functions of an OS

1. Process control
2. Memory management
3. Protection and security
4. User interface
5. File Management
6. Interrupts Handling

1. Process control

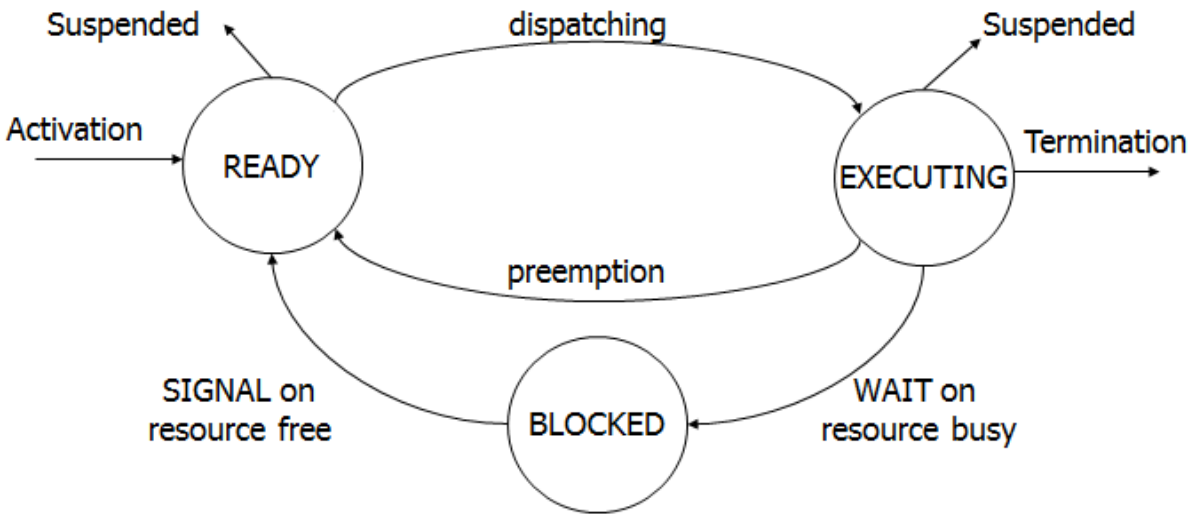
A process is a program in execution.



Pre-emption: the OS stops a program from continuing execution.

Dispatching: choosing a program to run by the CPU. The dispatcher is an OS module that selects the next process for execution.

Context switching: a context switch is a procedure that the CPU follows to change from one process to another. The term "context" refers to the data in the registers (memory cells inside the CPU).



States of a process:

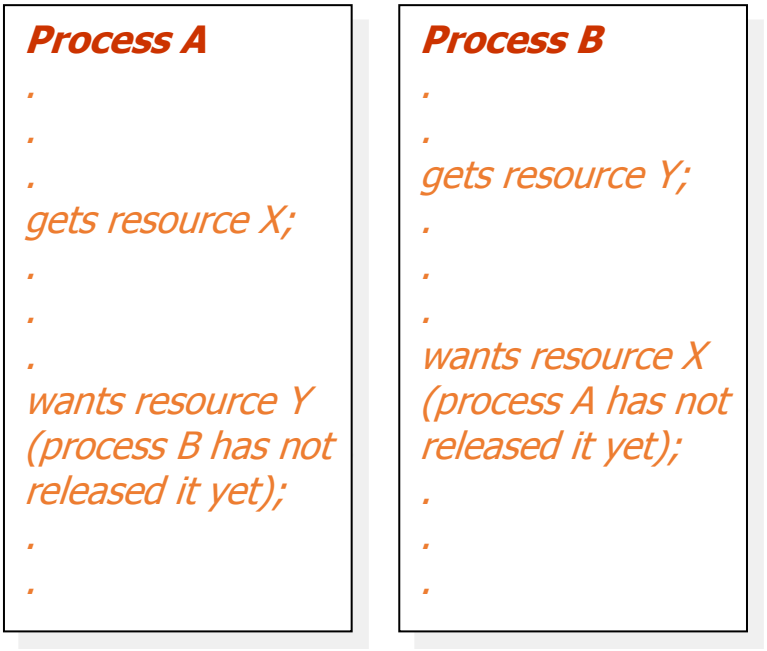
- Executing: is using the CPU.
- Ready: is waiting to use the CPU.
- Blocked: cannot use the CPU because of a blocking event.

Scheduling (the method how the processes share the CPU):

- First come first served (FCFS)
  - Holds processes in a queue.
  - Non-pre-emptive.
  - Used when programs are non-interactive.
  - Not used in modern OSs.
- Priority scheduling
  - Each process has a priority
  - Two versions:
    - Non-pre-emptive: a FCFS queue for each priority
    - Pre-emptive: time-sharing is applied but the higher-priority processes are executed first.
- Round robin
  - Choosing processes one after the other giving them a time-slice each. Then repeat again.
  - Suitable for time-sharing systems.

Deadlock

- Occurs when two processes are holding each other from proceeding.



There are four approaches to deal with deadlocks:

- Deadlock prevention: build a system so that deadlocks can never occur. This limits the systems' features.
- Deadlock avoidance: a request for a resource is considered and if it is concluded that its assignment can cause a deadlock the resource is not given. This is costly in overhead.
- Deadlock detention: system checks whether there are deadlocks and if it finds a deadlock are one or more processes are aborted. There is the cost of overhead.
- Ignore problem: they system makes no checks whatsoever. If a program is deadlocked its user will abort it and restart it.

2. Memory Management

Memory management refers to main memory management. Memory is a sequence of locations (memory cells). Each cell has a unique address (refer to top diagram pg. 11). It has direct (random) access.

Uni-programming: 1 program in RAM.

Multiprogramming: 2 or more programs in RAM

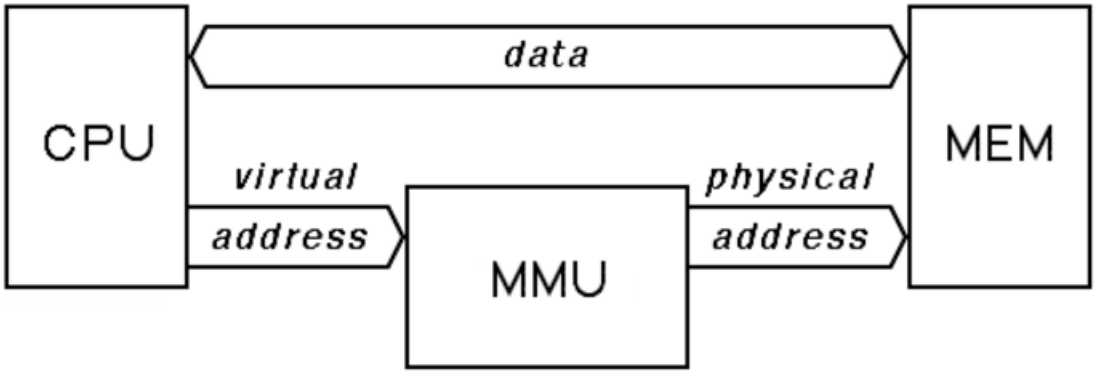
Refer to bottom diagram on pg. 11

Cache	Holds frequently used processes and/or data.

Main memory	Holds programs (processes) while they are being processed.
Secondary storage	Stores files permanently.

Memory map: describes what is found in memory and where (refer to diagram on page 12)

A logical address is a virtual address and can be viewed by the user. The logical address is generated by CPU during a program execution whereas, the physical address refers to a location in the memory unit. The physical address is computed by MMU (memory management unit).



Memory fragmentation: the occupied and non-occupied parts are divided into non-contiguous parts.

Compaction: removing the spaces between programs in memory. This is done by relocating programs so that they occupy contiguous spaces. See bottom diagram on page 13 and diagrams on pg. 14.

Process control block (PCB):

- Each process (that has not terminated) has a PCB.
- A PCB consists of information about the process e.g.
  - Process State - Running, blocked (waiting), etc.
  - Process ID, and parent process ID.
  - CPU registers and Program Counter (these need to be saved and restored when swapping processes in and out of the CPU).
  - Priority
  - Where, in memory, is the process.
  - CPU time consumed, account numbers, limits, etc.
  - Devices allocated, open file tables, etc.

The OS sees that processes do not reference memory locations belonging to other processes. See figure pg. 15.

Address references are checked at run time by hardware (software checks are too slow).

Virtual Memory

- Means simulating more RAM than actually exists.
- Computer can run larger programs and multiple programs concurrently.
- Running programs are held partly in the RAM and partly on secondary storage.
- RAM is divided in pages typically 4KB in size.
- Swapping means replacing the content in a page with other content from secondary storage.
- Advantage: second point
- Disadvantage: overhead (i.e. system is slowed)
- See diagram pg. 16

3. Protection and Security

Protection and security

Protection and security are very similar terms. But

- Protection: against non-malicious attacks
- Security: against malicious attacks

Goals of Protection and Security

- To prevent misuse (malicious or not)
    - Examples of misuse: theft of private or confidential information; unauthorised access; unauthorised modification or destruction of data; denial of service; virus (e.g. boot-sector computer virus, macro virus); worm; social engineering (i.e. means fooling trustworthy people into accidentally breaching security); Phishing (i.e. sending an innocent-looking e-mail or web site designed to fool people into revealing confidential information); Trojan Horse (a program that secretly performs some maliciousness in addition to its visible actions); spyware; Trap Door (is when a designer or a programmer deliberately inserts a security hole); logic bomb (software that will cause havoc on a particular date)
    - Measures: passwords, antivirus, firewalls, encryption, authentication (involves verifying the identity of the entity who transmitted a message), digital certificate (An attachment to an electronic message used for security purposes. The most common use of a digital certificate is to verify that a user sending a message is who he or she claims to be), biometrics (Fingerprint scanners, Palm readers, Retinal scanners, Voiceprint analysers, etc.); auditing, accounting etc.;
- To ensure that each shared resource is used only in accordance with system policies

    - Examples of misuse: theft of CPU cycles; tapping communication lines
    - Implemented by access rights (no-access, execute, read, write, print) etc.
- To ensure that errant programs cause the minimal amount of damage possible
    - Examples: (1) program jumps in the area of another program; stack or buffer overflow;
    - Implemented (1) hardware checks on jumps; (2) abort a program that causes the overflow and clear stack or buffer
- To ensure that the system is kept in the best environment (e.g. right temperature)
    - Implemented by air conditioners etc.
- To be prepared in case of disasters
    - Backups in a remote place

4. User Interfaces

User interface (UI):

- junction between the user and the computer
- most commonly UIs:
    - Command-Driven Interface (CLI): user types commands from the keyboard
    - Menu-driven interface: user selects option from menu
    - Graphical User Interface (GUI): also called WIMP. Stands for windows, icons, mouse (or menus), pointer.
    - Touch user interface
    - Gesture interface
    - Voice user interface

5. File Management

File management system:

- provides support to help users and applications create, delete, and use files

File:

- can be looked at as:
    - a sequence of bits (low-level perspective)
    - a stream of bytes (low-level perspective)
    - a structure of (high-level perspective):
        - words (e.g. a story)
        - pixels (e.g. photos)
        - records (e.g. a database table)
- files are divided into blocks (a block is a unit of storage both on disk and in main-memory buffers)
- file organisation and access (file of records)
    - serial files

        - not sorted
        - sequential
    - sequential
        - sorted on a key-field
    - indexed sequential
        - usually used sequentially but has index for direct access
        - index may be a complex tree structure, or a simple list
    - direct
        - random access
        - uses an index or other techniques
        - advantages:
            - faster if one needs to access one or few records
            - if a new record needs to be inserted only the index would need to be modified
        - disadvantage
            - index

Files are organised in folders in a tree structure. A pathname describes where a file is found by going inside folders.

File Sharing

- Two issues:
    - Determining access rights
    - Managing simultaneous access

The most common access rights (also called attributes) are:

- read: user can read and use the file, but can't change it
- write: user can modify, delete or add to the file
- execute: user can load and run the program, cannot change or copy it
- others:
    - append
    - delete

Access can be granted to different classes of users:

- individuals
- groups (a set of users, such as class members)
- all (includes everybody with access to the system - e.g., public files)

Unauthorised Access

- Some system administrators set up alerts to let them know when there is an unauthorized access attempt.
- Many secure systems may also lock an account that has had too many failed login attempts.
- A firewall is designed to prevent unauthorized access to or from a private network.
    - Firewalls can be implemented in both hardware and software, or a combination of both.

- Firewalls are frequently used to prevent unauthorized Internet users from accessing private networks connected to the Internet, especially intranets.
    - An intranet is a private network based on TCP/IP protocols i.e. it is a small private internet
    - All messages entering or leaving the intranet pass through the firewall, which examines each message and blocks those that do not meet the specified security criteria.

6. Interrupts Handling

What is an Interrupt?

- It is a signal to the processor emitted by hardware or software indicating an event that needs immediate attention.
- The processor responds by suspending its current activities, saving its state, and executing a small program called an interrupt handler (or interrupt service routine, ISR) to deal with the event.
- This interruption is temporary, and after the interrupt handler finishes, the processor resumes execution of another process.
- There are two types of interrupts:
    - Hardware interrupt
        - This is an electronic alerting signal sent to the processor from an external device, either a part of the computer itself such as a disk controller or an external peripheral.
        - Examples:
            - Pressing a key on the keyboard
            - Moving the mouse triggers hardware interrupts that cause the processor to read the keystroke or mouse position.
        - Unlike the software type (below), hardware interrupts are asynchronous and can occur in the middle of instruction execution.
        - The act of initiating a hardware interrupt is referred to as an interrupt request (IRQ).
    - Software interrupt
        - This is caused either by an exceptional condition in the processor itself, or a special instruction in the instruction set which causes an interrupt when it is executed.
        - Software interrupt instructions function similarly to subroutine calls and are used for a variety of purposes, such as:
            - To request services from low level system software such as device drivers.
            - To communicate with the disk controller to request data that be read or written to the disk.
            - Each interrupt has its own interrupt handler.

- Interrupts are a commonly used technique for computer multitasking, especially in real-time computing. Such a system is said to be interrupt-driven.

Polled and Vectored Interrupt

- A polled interrupt is a specific type of I/O interrupt that notifies the part of the computer containing the I/O interface that a device is ready to be read or otherwise handled but does not indicate which device. The interrupt controller must poll (send a signal out to) each device to determine which one made the request.
    - Polling is also called busy waiting.
    - This is generally not as efficient as the alternative to polling, interrupt-driven I/O.
- The alternative to a polled interrupt is a vectored interrupt, an interrupt signal that includes the identity of the device sending the interrupt signal.

Multiple Interrupts and Interrupt Priorities

- A multiple interrupt is an interrupt that interrupts an interrupt handler.
- Two techniques to handle multiple interrupts are the following:
    - Disable interrupts while an interrupt is being processed.
        - Advantage: solution is simple because interrupts are handled in strict sequential order.
        - Disadvantage: it does not take into account relative priority and time-critical needs.
    - Define priorities for interrupts and allow an interrupt of higher priority to interrupt an interrupt-handler of lower priority.

Interrupt Mask Register

- An interrupt mask is an internal switch setting that controls whether an interrupt can be processed or not.
- The mask is a bit that is turned on and off by the program.
- An interrupt mask register is a register holding a series of interrupt masks.